

AFFTC-PA-12672



Spectrum Savings from High Performance Recording and Playback Onboard the Test Article

Mark Wigent

**AIR FORCE FLIGHT TEST CENTER
EDWARDS AFB, CA**

2/20/13

**A
F
F
T
C**

Approved for public release; distribution is unlimited.

**AIR FORCE FLIGHT TEST CENTER
EDWARDS AIR FORCE BASE, CALIFORNIA
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE**

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (20-02-2013)		2. REPORT TYPE Technical		3. DATES COVERED (From - To) 3/12 -- 10/12	
4. TITLE AND SUBTITLE Spectrum Savings from High Performance Recording and Playback Onboard the Test Article		5a. CONTRACT NUMBER W900KK-11-C-0028			
		5b. GRANT NUMBER			
		5c. PROGRAM ELEMENT NUMBER			
6. AUTHOR(S) Mark Wigent		5d. PROJECT NUMBER			
		5e. TASK NUMBER			
		5f. WORK UNIT NUMBER			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) SCIENCE APPLICATIONS INTERNATIONAL CORP PIERCE MARTIN, 1710 SAIC DR, MCLEAN VA 22102-3703		8. PERFORMING ORGANIZATION REPORT NUMBER 412TW-PA-12672			
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Tom Young, EA		10. SPONSOR/MONITOR'S ACRONYM(S) N/A			
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)			
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release A: distribution is unlimited.					
13. SUPPLEMENTARY NOTES CA: Air Force Flight Test Center Edwards AFB CA CC: 012100					
14. ABSTRACT Network data recorder that provides instantaneous access to recorded flight data without interrupting ongoing recording Parametric-level data retrieval for backfill of data dropouts and ad hoc requests Need for spectrally efficient retransmission. Recording data at high rates while at the same time allowing for concurrent, rapid and selective data retrieval Developing approaches to data indexing, storage, and retrieval that optimize overall system performance Developing ways to retrieve data based on complex criteria, including values of the stored data itself and derived values such as minimum, maximum, average within a time interval Parallelizing storage and retrieval mechanisms to enhance performance					
15. SUBJECT TERMS Spectrum Savings from High Performance Recording and Playback Onboard the Test Article, EQDR, Enhanced Query Data Recorder, Spectrum Efficient Technology, Telemetry					
16. SECURITY CLASSIFICATION OF: Unclassified		17. LIMITATION OF ABSTRACT None	18. NUMBER OF PAGES 12	19a. NAME OF RESPONSIBLE PERSON 412 TENG/EN (Tech Pubs)	
a. REPORT Unclassified	b. ABSTRACT Unclassified			c. THIS PAGE Unclassified	19b. TELEPHONE NUMBER (include area code) 661-277-8615

SPECTRUM SAVINGS FROM HIGH PERFORMANCE NETWORK RECORDING AND PLAYBACK ONBOARD THE TEST ARTICLE

Mark A. Wigent
SAIC

Dr. Andrea M. Mazzario
Kauai Software Solutions

ABSTRACT

The Test Resource Management Center's (TRMC) Spectrum Efficient Technologies (SET) S&T program is sponsoring development of the Enhanced Query Data Recorder (EQDR), a network flight recorder that is intended to meet the future needs of the networked telemetry environment. EQDR is designed to support the "fetch" of recorded test data during a test without interrupting the ongoing recording of data from the test article vehicle network.

The key benefits of the network data recorder as implemented in EQDR are increased flexibility and efficiency of test in an environment with increasing demands on spectrum available for telemetered data. EQDR enables retrieval of individual recorded parameters on an as-needed basis. Having the flexibility to send data only when it is required rather than throughout the duration of the test significantly increases the efficiency with which limited spectrum resources are used. EQDR enables parametric-level data retrieval, based not only on time interval and data source, but also on the content of the recorded data messages. EQDR enables selective, efficient retrieval of individual parameters using indexes derived from the actual values of recorded data.

This paper describes the design of EQDR and the benefits of selective data storage and retrieval in the application of networked telemetry. In addition it describes the performance of the EQDR in terms of data recording and data retrieval rates when implemented on single board computers designed for use in the aeronautical test environment with size, weight, and power constraints.

KEY WORDS

Network data recorder, flight recorder, iNET, spectrum efficiency, enhanced query data recorder

INTRODUCTION

As described in the *iNET Needs Discernment Report* (2004), the T&E community has identified a number of use cases that call for the ability to “fetch” data on demand from a test article, without interrupting the ongoing recording process. This use case is one of the fundamental motivators for the capabilities being developed by the Central Test and Evaluation Investment Program’s (CTEIP) iNET project. Also in response to this need, the Test Resource Management Center’s (TRMC) Spectrum Efficient Technologies (SET) T&E S&T program is sponsoring development of the Enhanced Query Data Recorder (EQDR), a network data recorder that is intended to meet the future needs of the networked telemetry environment. EQDR is a network data recorder that enables retrieval of individual recorded parameters on an as-needed basis, in response to queries from the ground, and supports concurrent recording of all data on the test article bus.

The key benefits of the network data recorder as implemented in EQDR are increased flexibility and efficiency of test in an environment with increasing demands on spectrum available for telemetered data. Having the flexibility to send data to ground only when it is required, rather than throughout the duration of the test, significantly increases the efficiency with which limited spectrum resources are used. EQDR enables parametric-level data retrieval, based not only on time interval and data source, but also on the content and characteristics of the recorded data themselves. EQDR enables selective, efficient retrieval of individual parameters using indexes derived from the actual values of recorded data.

This paper will describe the design of EQDR and the benefits of selective data storage and retrieval in the application of networked telemetry. In addition it describes the performance of the EQDR in terms of data recording and data retrieval rates when implemented on single board computers designed for use in the aeronautical test environment with size, weight, and power constraints.

T&E BENEFIT

The capability to pull data from an onboard flight recorder and transmit those data to the ground during flight test ensures that the test engineer can receive all required test data. This benefit is significant to the T&E community. The need to fetch data from the test article may arise for several reasons, including data dropouts and changes in test mode, such as inter-maneuver periods and non-nominal phases of flight.

Telemetry data dropouts are unavoidable in T&E, especially in the aeronautical test community where aircraft maneuvers result in the degradation of received signals. In addition, various radio frequency (RF) environments encountered during a flight test (e.g., flight line, flight path over the ocean or a dry lake bed, or hilly terrain) can cause obstruction or multipath, fading, or line-of-sight signal degradation, also resulting in data dropouts. During these dropout periods, critical data are lost, and because certain short duration dropouts are not always detected during the test, erroneous interpretation of real-time data can result. Current flight recorders do not provide a mechanism to retrieve data to fill in these data dropouts. EQDR was designed to respond to requests from the

ground, via a two-way communications link, to replay any individual measurands that were previously recorded. As the retransmitted must compete with the ongoing transmission of measured data, the EQDR is highly selective and can replay only the specific parameters that are requested. In addition, the replay rate can be slowed, if the user desires, so the retransmitted data does not excessively compete for bandwidth with the real-time telemetry stream.

The amount of data generated on board the test article exceeds by at least an order of magnitude the amount of spectrum available to support real-time transmission of onboard sensed parameters. As a result, test engineers must choose to send only the most critical data to mission control facilities in real time, while recording the remainder of data on board for post-mission analysis. If an unanticipated or emergency mode occurs during a test event, the test engineer in mission control may want to view additional recorded data related to the observed anomaly that was recorded but not originally transmitted to ground because of spectrum limitations. Doing so may be critical in real time to ensure safety of flight or to preclude unnecessary termination of flight test. Ad hoc requests can also be made during inter-maneuver periods to ensure that the test objectives have been met. Test engineers can issue ad hoc requests of key data for quick-look analysis to ensure that the right data was collected before moving on to the next test point. Current flight recorders have no mechanism for retrieving specific measurements from the flight recorder for analysis during flight test.

EXISTING TECHNOLOGY

The overall benefits to the range community include more efficient use of available spectrum, increased safety of flight, more efficient use of range resources, and reduced cost of test. Existing IRIG 106 Chapter 10-compliant recorder technology does not support these requirements.

The digital recording standards for currently available recorders are described thoroughly in the RCC IRIG 106 Chapter 10. Chapter 10 defines many characteristics of the recorder, including how data are stored. Based on this standard, flight recorders store data in blocks as a function of time interval and channel only. Page 10-21 of RCC document 106-07, illustrates the directory structure to which a Chapter 10-compliant recorder must adhere. This directory structure contains only the time the file was created and the time the file was closed, thus specifying the interval during which the file was recorded.

Further, information about which channels were used in recording the file is contained in the data recording structure illustrated in pages 10-27 of the same RCC document. Each data file consists of a number of data packets interspersed with time data packets. Each data packet should contain data for a time interval of no more than 100 msec, and the time packets should be inserted into the data file at least once each second.

Each packet has at least one packet header, which must include the channel identification (ID), which is the channel from which the data in the packet were recorded. The packet

also has a Data Type field that specifies the format of the data in the packet, such as PCM. These data types are enumerated in Table 10-7 of the IRIG 106-07 document. This format contains no information related to which measurands are contained in the data packet.

In conclusion, the definition of the Chapter 10 directory allows retrieval of a file based on the time interval of the recording only. By scanning the packet headers in the file, existing recorder software could retrieve the packets that contain the data for a set of desired channel IDs over a given time interval. Chapter 10 requires that each packet cover no more than 100 msec, so the time discrimination will have at least that accuracy, but the packet has no way to filter data with granularity higher than the number of channel IDs. Moreover, the packet cannot filter data based on the values of the data themselves.

S&T CHALLENGES

A corollary requirement to the need to fetch data on demand is the need to do so with maximum spectrum efficiency. Recorded data that are transmitted to the ground during flight must be inserted into the transmitted data stream and, as a result, compete with other data for very limited spectrum. Therefore, it is an inherent requirement that the flight recorder support *efficient* retrieval of specific recorded parameters. If the recorder responds to data requests with large blocks of extraneous data, along with data that are truly needed, then the retransmission of data represents a costly and inefficient use of spectrum. For spectrum efficiency, it is imperative that the network data recorder be able to store, and retrieve, data at a parametric level.

The key characteristic for a successful network flight recorder is the ability to record data at a sufficient rate in terms of megabits per second, while at the same time recording data at a sufficient level of granularity and indexing to enable very selective data retrieval for retransmission. A simplified view of the technical trade-space is that storing data in large blocks eases the burden on the recorder's processor and input/output (I/O) mechanisms, enabling high recording rates, whereas storing data in smaller increments facilitates retrieval and retransmission of data at the cost of lowering recording rate. Existing recorders sit at one extreme of this trade-space. Playback of selected data during recording is not a required feature of traditional data recorders because traditional PCM telemetry does not support the retransmission of recorded measurands. Traditional flight recorders record large blocks of data with only minimal information about their content (typically, only time blocks in which the data was generated and channel information). With only that information at hand, it is impossible to select only specific measurands for retransmission.

The S&T challenge addressed by the EQDR technology lies in developing a network recorder that stores data in a way that allows retrieval of individual measured parameters quickly and efficiently and, at the same time, meet requirements for recording rate on the vehicle network. EQDR advances the state of the art in recorder technology by providing full support of the iNET Telemetry Network System (TmNS) message format, enabling

concurrent recording with data fetch-retransmit capability, and enabling parametric-level data retrieval, based not only on time interval and data source, but also on the contents of the recorded data.

CONCEPT OF OPERATIONS

The following test scenario describes key features of the EQDR and how the end-user tester will benefit from using it. The test scenario includes three instances in which selective retrieval and retransmission of data improve test efficiency and safety of flight: data dropouts, non-nominal modes of operation, and inter-maneuver periods.

The new Fighter-X aircraft is under test and executing the first of several test cards focused on testing various systems of the aircraft. As part of the first test card, the aircraft is slated to execute a series of high G maneuvers to test its airframe performance. During one of these maneuvers, a data dropout occurs in the telemetry data received by the ground. This dropout occurs in the middle of that maneuver, during the time at which the greatest stress is placed on the airframe. Telemetered data from the strain gauges on the wing of the aircraft appear normal and within acceptable bounds for safe flight before and after the maneuver. However, data generated during the maneuver are missing. The onboard data recorder has recorded these data, and they will be available for post-mission analysis after the aircraft has landed. Nonetheless, test engineers on the ground want to view data immediately to ensure that readings on the wing strain gauges stay within the safe operating range of the aircraft.

The aircraft is equipped with iNET system, which includes an Ethernet vehicle network, TmNS message types, and two-way radios, and the EQDR. Upon observing the data dropout, test engineers quickly send the test article a request for retransmission of the missing strain gauge data. The EQDR receives the request.

Because the aircraft is going through a series of important tests, bandwidth available for retransmission of data is limited. The ground engineers need to receive the strain gauge data quickly but without interruption in the transmission of other parameters; therefore, the EQDR must be selective in the data it retrieves. Because the EQDR has preprocessed all TmNS data packets and recorded measurands within a database, the EQDR is able to quickly retrieve the strain gauge data, and *only* the strain gauge data, during the time interval in which the data was originally lost. The EQDR then reconstructs those data into TmNS messages and publishes them to the test vehicle network, and they are incorporated into the telemetry stream by the test article transmitter. The impact on the existing transmission is minimal because only critical strain gauge data are included in the retransmission. Had the regenerated TmNS packets from the EQDR included extraneous data from the original TmNS messages containing the strain gauge data, this retransmission would not have been possible without interrupting the ongoing transmission of data. The engineers verify that, during the data dropout, strain on the airframe remained within acceptable levels for safe flight. The test continues, and Fighter-X finishes the series of maneuvers in that test card.

With the maneuvers complete, the test article proceeds to a new test card. The aircraft executes a number of maneuvers successfully, but after some time, one of the mission control engineers observes that fuel pressure on board the aircraft drops below its nominal operating range. Test engineers would like to analyze a small amount of historical data to assess the situation. From mission control, they issue a control request to the EQDR, initiating a retrieval of oil pressure and oil temperature data, which were not previously sent to ground, for the last two minutes immediately preceding the time at which fuel pressure dropped below nominal level.

As was the case for the data dropout, the EQDR is able to quickly retrieve those specific parameters, which originally were associated with several different TmNS message sources. Had raw, unprocessed TmNS messages been recorded, the retrieval would have contained a great deal of extraneous data. However, the EQDR already has processed the TmNS data and used a DBMS to store them so they can be retrieved quickly and selectively. EQDR reconstructs these data into a TmNS message and publishes them to the vehicle network where they are received and sent to ground. Test engineers observe a steady decline of oil pressure and increase in oil temperature during the two-minute window before the drop in fuel pressure. They hypothesize that the drop in oil and fuel pressure was related to the maneuver that was being conducted at that time. Now that the maneuver is over, all three parameters—oil pressure, fuel pressure, and oil temperature—have returned to normal.

At this point, the aircraft has entered an inter-maneuver period. Mission control engineers are still concerned about the drop in fuel pressure, and they would like to do additional analysis before continuing on to the next test. They want to verify that the drop in oil pressure was occurring only during that specific maneuver or to determine whether, at any time during other maneuvers, this same phenomenon occurred. They know that fuel pressure previously had not dropped below nominal because they had been monitoring the pressure in real time, but because oil pressure and oil temperature were not in the predefined measurement list, they were not watching those parameters. At the same time, they would like to look at several other parameters that would help corroborate their theory on what caused the drop in fuel pressure. Rather than sending a control request to the EQDR to retransmit every parameter for the last hour of test time, they issue a query to search all recorded data for instances in which oil pressure, oil temperature, and other parameters of interest went above specific maximum values or below minimum values. The EQDR efficiently retrieves all periods of flight in which those criteria are met, and these data sets are sent to the ground.

On the ground, mission control receives the data. The EQDR query capability allows the test engineers to review only the data of interest. Rather than having to sort through an hour's worth of test data, they receive four minutes of test data, as during all other periods of time, all parameters were within nominal range. They learn that the low oil pressure occurred only in the four-minute period before the drop in fuel pressure. This observation confirms their hypothesis that the drop in oil and fuel pressure correlates to the specific maneuvers being conducted. Test engineers decide that it is safe to continue

with the remaining scheduled tests, as none of them will reproduce the conditions that caused the fuel pressure drop. The aircraft proceeds to execute the next set of tests.

DESIGN AND IMPLEMENTATION

The EQDR system comprises two major components. First is EQDR Recorder module, which is the actual flight recorder that sits on the test article. This consists of one or more processor boards that run the underlying system software, solid-state drives (SSDs) for data storage, and an Ethernet interface. The second major component of the EQDR system resides on the ground and consists of a user interface and application code that processes retransmission requests and queries. The EQDR software and underlying database execute within a Windows 7 environment, and data is recorded on SSDs. The underlying database is implemented using MySQL. Figure 1 illustrates the major components of the EQDR architecture.

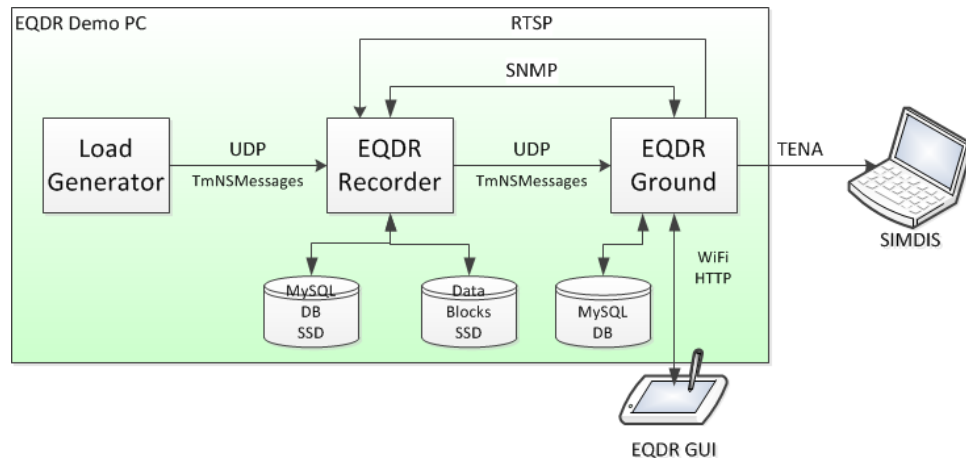


Figure 1 EQDR System Components

EMBEDDED SYSTEM PERFORMANCE

Fundamentally, the EQDR is able to efficiently retrieve recorded data at a parametric level because it has pre-processed all data on the test article as it is being recorded. During this pre-processing step, the EQDR generates metadata about all parameters as they are being recorded, and it uses this metadata as indexes for rapid, selective (e.g. spectrally efficient) retrieval at a later time. The essential trade-off of this design is between CPU cycles and spectrum utilization during retransmission, and given the sustained rapid growth in microprocessor capability versus the continuing encroachment on DoD telemetry spectrum, this is an efficient trade to make. Modern multi-core processors are highly capable, yet they are available in packages and form factors that meet size, weight, and power constraints of the aeronautical test environment.

We tested the performance of the EQDR on a reference system described in more detail in Figure 2 below. Key performance requirements for the EQDR included record rate of 250 Mbits/sec with a concurrent parametric level data replay rate of 30 Mbits/sec on the reference system without exceeding 50% CPU utilization. In addition to measuring CPU utilization, we measured the time elapsed between the reception of the TmNS Data

Messages and the writing of the metadata rows corresponding to their data blocks in the MySQL database. This is effectively the time at which the recorded data are available for retransmission. CPU and Memory utilization were collected using the Windows Perfmon tool.

When considering these elapsed times we must keep in mind that the frequency by which the data blocks are written to the SSD is configurable, and is set to 100 msec in our tests. This introduces a uniformly distributed random variable with a mean delay of 50 msec to our latency measurements.

Test System Configuration

- Intel i7 610 @ 2.53 GHz, 4 Threads
- 8 GB RAM
- Windows 7 Professional SP1 64 bits
- Java 7.0.3 64 bits

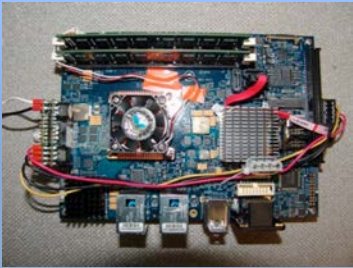


Figure 2 Test Configuration

We executed the performance tests using 1 to n TmNS data sources, each producing 20 Mbits/sec of data. We continually added data sources while measuring CPU loading and latency. Below we present results of two key points: 260 Mbit/sec (13 data sources), 460 Mbits/sec (23 data sources). They indicate that the EQDR can process the target load of 260 Mbits/sec on the reference system (Intel i7 610 @ 2.53 GHz, 4 Threads) using less than 30% CPU. Maximum sustainable recording on that board (defined by a CPU utilization of less than 50%) is 460 Mbits/sec. In addition, retransmissions as high as 60 Mbits/sec can be performed while recording with minimal impact on CPU utilization and recording latencies.

Target Recording Load: 260 Mbits/sec

This is the threshold recording rate of the EQDR on the reference system without exceeding 50% CPU load. CPU utilization during this test is shown in Table 1 below.

	CPU Load
System Total avg.	27.8%
EQDR Recorder avg.	17.7%
MySQL avg.	3.9%
EQDR Total avg.	21.6%

Table 1 CPU Utilization with 260 Mbits/sec Load

The difference between the total System CPU (27.8% in this test) and the one used by the EQDR processes (EQDR recorder and MySQL, total of 21.6% in this test) is due to the various windows system functions, mainly related to elements of the UDP network traffic

processing. This measurement shows that we are well under the total target CPU load of 50%, even when using the relatively modest CPU of the reference system.

Latency between the time TmNS Data Messages are received by EQDR and the time their data are available for queries or retransmissions is shown in Figure 3 below. This latency distribution has been measured with the EQDR configured to buffer the MySQL rows insertion requests for 100 msec, which adds an average of 50 msec of queuing delay to our measurements.

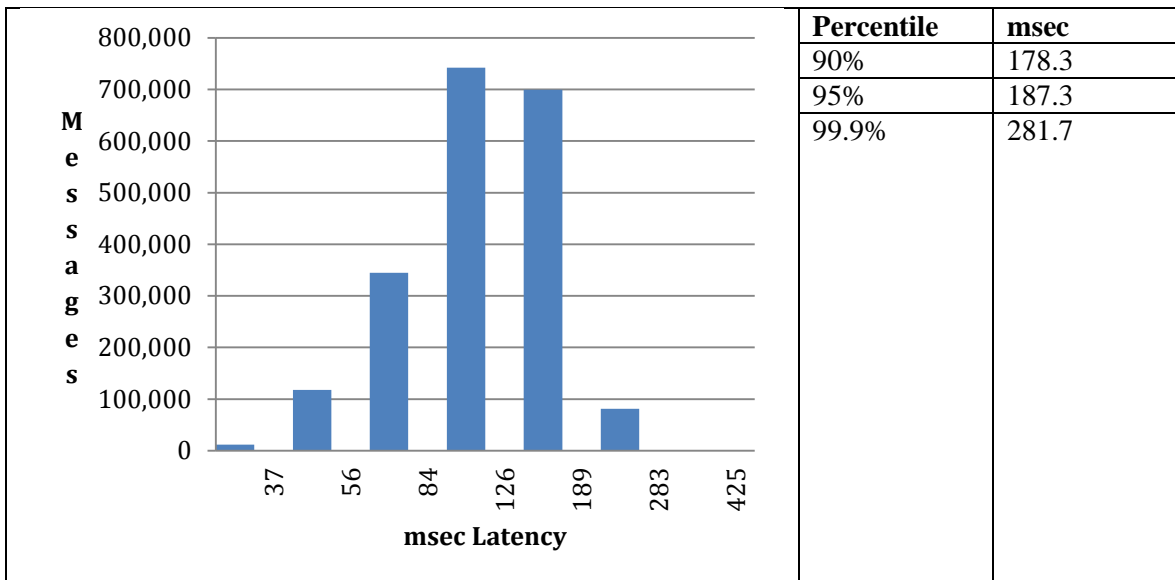


Figure 3 System latency while recording 260 Mbts/sec

Target Recording Load: 460 Mbts/sec

CPU utilization with 23 data sources is shown below in Table 2. This is the maximum recording rate that the EQDR can perform on the embedded computer within the 50% CPU load range.

	CPU Load
System Total avg.	50.1%
EQDR Recorder avg.	35.6%
MySQL avg.	7.0%
EQDR Total avg.	42.6%

Table 2 CPU utilization while recording 460 Mbts/sec

EQDR system latency at a recording rate of 460 Mbts/sec, with an average queuing delay of 50 msec is shown below in Figure 4.

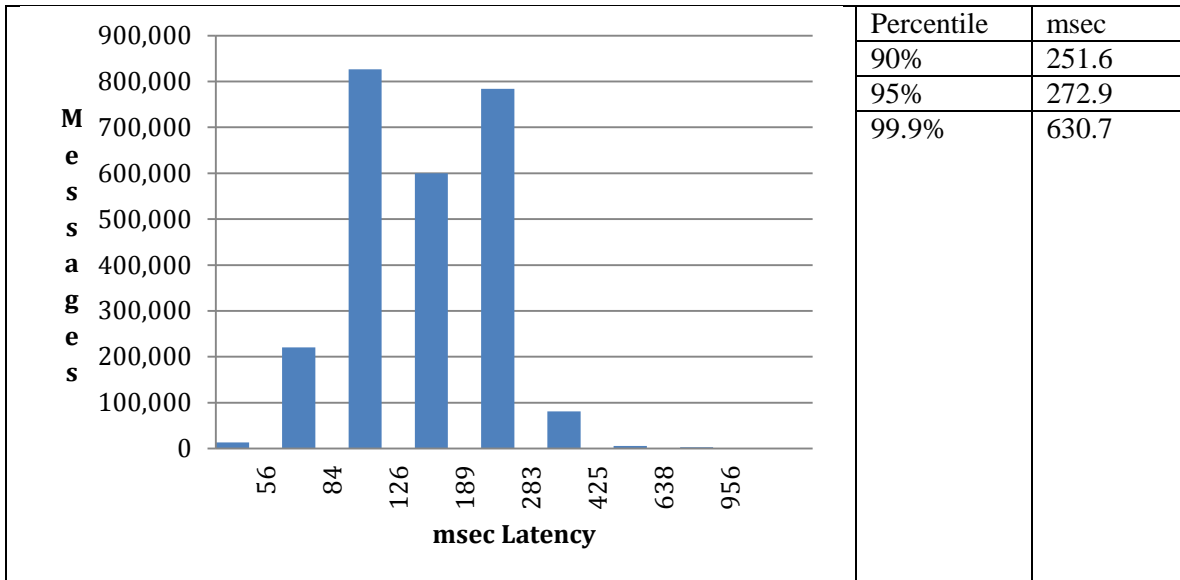


Figure 4 System latency while recording 460 Mbits/sec

Retransmissions while recording the Target load (260 Mbits/sec)

We performed a number of tests to measure the impact data retransmission upon CPU utilization, while recording at the target load (260 Mbits/sec). For each retransmission test we include the CPU utilization and the latency of writing the metadata rows to the database during the retransmission. It turns out that for most tests the overhead of the retransmission is so small that the difference in CPU load and especially latencies between the simple recording and recording with retransmission falls within the normal fluctuation of the measurements.

	250 Kbits/sec Retransmission	5 Mbits/sec Retransmission	20 Mbits/sec Retransmission	60 Mbits/sec Retransmission
System Total avg. CPU Load	26.6%	27.4%	28.2%	29.3%
EQDR Recorder avg. CPU Load	16.2%	17.6%	18.0%	18.1%
MySQL avg. CPU Load	3.9%	3.9%	3.9%	4.0%
EQDR Total avg. CPU Load	20.1%	21.5%	21.9%	22.1%
Percentile	msec	msec	msec	msec
90%	176.9	177.3	179.6	176.8
95%	185.9	186.6	187.0	185.0
99.9%	344.0	283.2	283.1	279.9

Table 3 CPU utilization during retransmission

SUMMARY

The key benefits of the network data recorder as implemented in EQDR are increased flexibility and efficiency of test in an environment with increasing demands on spectrum available for telemetered data. EQDR enables retrieval of individual recorded parameters on an as-needed basis.